Route Design for Freight Trip Based on an Enhanced Greedy Algorithm

Mingjun Yan(✉)

Kang Chiao International School, Huaqiao, Kunshan 215332, China
1924262597@qq.com

Abstract. As a result of the COVID-19 pandemic, online shopping, due to its convenience and ever-improving digital experience, is rapidly becoming the preferred method of purchasing for by an increasing number of individuals. With the efficiency of purchasing improving, the growing number of orders puts more pressure on freight. Consequently, the deliverymen must deliver more packages than they did before and face a substantial rise in their workload. Arguably, it is crucial to re-construct and optimize the delivery routes to reduce the workload of the delivery staff, increase customer satisfaction, and lower the company's expenses. This research aims present an enhanced greedy algorithm for designing more efficient delivery routes to reduce delivery distance and delivery time. The author generates several edges to simulate city-to-city routes and utilize C++ software as a tool to apply an enhanced greedy algorithm to determine the optimal path for delivery. It can be concluded from the experiment that the paths built with the enhanced greedy algorithm provide optimal results compared to those generated with the greedy algorithm that is not restricted by factors like time and number of packages. The end results of lowering the delivery time of packages and decreasing the distance of delivering can demonstrate the effectiveness of the enhanced greedy algorithm in express delivery path planning.

Keywords: Greedy algorithm · Route design · Freight

1 Introduction

Nowadays, with fast and continuous growth in logistics demand, logical physical route design has become even more vital. Logistics companies need to reduce the workload of delivery and reduce the expenses of the organization by rational route planning to achieve the least possible delivery time. Many scholars have developed numerous strategies for planning the shortest route. For example, the O(E) Time Shortest Path Algorithm which calculates efficient path between two points source and destination instead of traversing all the nodes [1], the Dijkstra Algorithm which discovers the shortest pathways between nodes in a network by traversing the graph [2] and Ant Colony Optimization Algorithm which is a flexible algorithmic optimization strategy based on ant behavior observation [3]. In recent years, many scholars have also proposed improvements to these methods. For example, YiBin Nie, MingJun Tan, Gang Liu, and Lu Ma improved the network

Y. Chen et al. (Eds.): ICMETSS 2022, ASSEHR 693, pp. 115–120, 2023.
https://doi.org/10.2991/978-2-494069-45-9_15

nodes, database, and heuristic functions in the traditional A* algorithm, thus enhancing the speed of the A* algorithm for planning paths [4]. Nengchuang Xu improved the Ant Colony Algorithm by searching the next node randomly for the probabilistic selection formula, thus improving the path planning speed [5]. Wenbin Sun and Jiang Wang (2016) significantly improved the path planning speed of the genetic algorithm by using a multi-strategy optimal solution method [4, 6]. There is no doubt that these algorithms are extraordinarily effective for path planning. However, in practical applications in the logistics business, the algorithms must account for the actual delivery time of the goods, the unloading volume of the distribution sites during the delivery process, and other elements that these better approaches do not examine. In this research, the author proposes to enhance the Greedy Algorithm for planning the local optimal route for logistics distribution.

2 Greedy Algorithm

2.1 Principle of the Greedy Algorithm

Each stage of the greedy algorithm has a locally optimum decision, and we can select whichever decision that seems best at the time and then solve the subproblems that occur later. The decisions made by a greedy algorithm might not be the global best option [5].

2.2 Conventional Approach of Greedy Algorithm in Designing Routes in Logistics

Changxian Zhao and Muyun Fang (2020) use the greedy method to determine the path between the starting and final distribution stations that requires the least amount of time or travel [7]. They recorded the distance and time spent at the delivery location, traversed the recorded set, and chose the delivery point with an existing path. Then, they chose the distribution station with the smallest weight among the other distribution stations and added it to the collection of acquired paths. Lastly, they added the distribution station that did not obtain the shortest path. Finally, the shortest path of logistics distribution can be identified.

2.3 Improvements of the Greedy Algorithm in Logistic Distribution

In the process of determining the shortest route, prior researchers have rarely addressed the delivery volume at the delivery site, the amount of time the delivery object has been stored at the delivery point, the amount of time it takes to reach the consumer. In this paper, the author used the greedy algorithm as a starting point and add the number of objects to be delivered at the delivery point and the time of the objects to be delivered as constraints to the conventional route planning in order to plan a more practical path for the real logistics distribution. The author assigned weights to the logistics vehicle load, delivery point delivery, and storage time of the to-be-delivered objects. The ratio between the delivery volume and the storage time of the objects is then recorded as a new distribution efficiency weight value. These weights are used as constraints for path planning.

3 New Algorithm Analysis

Based on the preceding design, each distribution point is set as a node and the traffic route between distribution points is set as an edge, and the distribution distance is assigned a value, while the distribution efficiency (distribution volume and dwell time of items to be distributed) is assigned a value, and the greedy algorithm is used to calculate the optimal path for each distribution, assuming the load capacity of logistics vehicles is not exceeded. In the algorithm, di is the path length, npi is the number of objects to be delivered at each distribution point, ti is the time for delivery of objects to be delivered at each distribution point, and npc is the maximum cargo capacity of the delivery vehicle. There are n distribution nodes in the algorithm and each node is given a marker value Bi to mark whether the node has been delivered or not. The initial value of Bi is false. First, the distribution nodes are sorted in descending order by traversing the distribution efficiency of all nodes. The graph traverse starts with the first node after sorting, which is the first place where the logistics vehicle goes after loading the distribution goods. The delivery vehicle's load npc and Bi are the termination conditions. When each node is finished, the value of npc is the initial value minus npi, and Bi is true. When the npc of the finished vehicle is positive and any node's Bi is false, the traverse continues. If the npc becomes negative when the current node is delivered, the current delivery is stopped, and this node i is the second starting node, Bi is marked as false, and npc is reset. If the traverse requirement is met, the most efficient node at the top of each traverse is chosen, the weights di of multiple paths corresponding to multiple nodes are extracted, and the paths are calculated using the greedy algorithm. In the traverse, each node is initially examined to determine if it is connected to the root node. If there is no path connecting the initial location, the next most efficient node is chosen, and so on, until the node connected to the starting point is identified. The total length of the existing paths is denoted by dtotal, and each time a path is walked. The selected nodes are visited, the weight di of each node connected to the start point is recorded, and the node with the smallest weight di and a high efficiency is connected to the starting point, and the sum is recorded as dtotal. This node next in the list becomes the next beginning point and after loading the goods, the logistics vehicle will start the next delivery from this point, followed by the next most efficient node that meets the traverse criterion among the remaining unselected nodes. The remaining unselected nodes are then added to the next traverse set along with the next more efficient node that satisfies the traverse criteria. Finally, dtotal is the total path length and the order of the nodes is the path sequence.

3.1 Design of the Experiment

There are 7 distribution points in a city. On a certain day, a freight company has to send a truck to deliver the items that need to be delivered to each distribution point. The maximum load capacity of each vehicle from each starting point, npc, is 5000 pieces of goods. In the distribution process, the transport road is clear, there is no traffic jam, and the vehicle does not experience a flat tire or other driving-related difficulties. The number of objects to be delivered at each distribution point npi, the time for delivery of objects to be delivered at each distribution point, ti, the distance di of the 7 distribution points are showed in Table 1.

Table 1. Data of experiment

	1st	2nd	3rd	4th	5th	6th	7th
di	8	11,7	9	6	17	15	12,13
npi	1250	3500	700	3000	2000	1000	600
ti	13h	26h	5h	30h	40h	22h	15h
npi/ti (in integer)	96	134	140	100	50	45	40

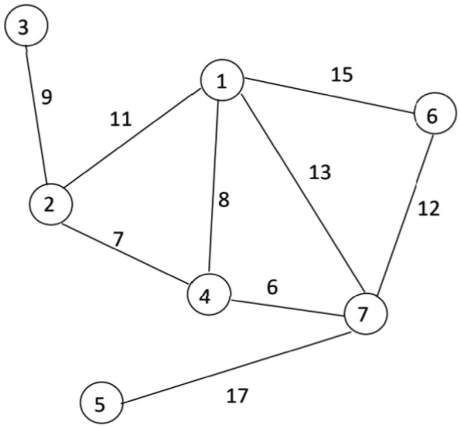

Fig. 1. Visual illustration of the experiment

The figure of the designed experiment is shown in Fig. 1.

3.2 Analysis of the Experiment

First, the distribution efficiency (di/ti) of each node is calculated, then the distribution efficiency is organized in descending order, and the node 2 with the highest distribution efficiency is identified; hence, node 3 is chosen as the starting point for distribution. The remaining nodes in order are node 2, node 4, node 1, node 5, node 6, and node 7. When satisfying the condition that the distribution volume npi is less than npc, the nodes that can be distributed are nodes 3 and nodes2. Therefore, dtotal is 9 at this time and the distribution order is 3→2. Continue traversing from node 2, the following nodes are nodes 4 and nodes 1. dtotal here becomes 16 when node 4 is delivered, and, after node 1 is delivered, dtotal equals 24. The sequence of delivery is 3→2→4→1 at this stage. The remaining nodes in order are node 5, node 6 and node 7. As nodes 6 and nodes 7 both have connections to node 1, while node 6's deliver efficiency is higher, node 6 is chosen as the next node and the node after it would be node 7. dtotal would be 51 and the order of delivery would be 3→2→4→1→6→7. The last node node 5 is then added to the sequence. so At the end of the experiment, a delivery route 3→2→4→1→6→7 with a total distance of 68 is found.

3.3 Comparison with Greedy Algorithm Without Constraints

When starting with node 3, the greedy algorithm without constraints gives an order of 3→2→4→1→7→6→7→5. This is because the algorithm only considers the weight of the distance, and at the end the dtotal of the planned route is 78, which is much greater than the dtotal calculated by the enhanced greedy method, meaning that it would take more distances for delivers to travel and would take more time to deliver the packages to consumers. Thus, effectiveness of the enhanced greedy algorithm can be determined by comparison.

4 Conclusion

In order to optimize the problems of long distribution distance and long distribution time in the actual logistics distribution route planning, this paper improves the Greedy Algorithm by adding the constraints of time and distribution volume and by adding the distribution efficiency (the ratio of distribution volume and distribution distance) in the route planning in addition to the conventional route distance weight variables, so as to give the greedy optimal solution in the local range and minimize the distribution distance and distribution time. By designing a program to simulate the distribution of inter-city logistics, the improved Greedy Algorithm was compared with the unimproved Greedy Algorithm in terms of the time spent on the path, and finally the improved algorithm optimized the length of the distance and the time spent on the distribution, proving that the improved algorithm did optimize the original algorithm. However, the improved Greedy Algorithm in this paper is only a locally optimal solution under the condition of satisfying the optimal value, so the solution obtained is not necessarily the global optimal solution. At the same time, the distribution volume of the logistics vehicles in this paper is not optimized to the extreme. In this paper, the distribution vehicles are rarely fully loaded for distribution when the amount needed to be distributed at the next distribution point is greater than the remaining amount that the vehicle can load. The vehicle will stop distribution and turn to redistribute from the next point, which will be wasteful for the amount of cargo carried.

References

1. Qureshi, Muhammad Aasim. Dr. Fadzil B. Hassan, Sohail Safdar, Rehan Akbar (2009). A O (E) Time Shortest Path Algorithm for Non Negative Weighted Undirected Graphs. International Journal of Computer Science and Information Security, IJCSIS, Vol. 6, No. 1, pp. 040–046.
2. Cormen, Thomas H.; Leiserson, Charles E.; Rivest, Ronald L.; Stein, Clifford (2001). "16 Greedy Algorithms". Introduction To Algorithms. MIT Press. pp. 370.
3. Glabowski, M., Musznicki, B., Nowak, P., & Zwierzykowski, P. (2012). Shortest path problem solving based on ant colony optimization metaheuristic. Image Processing & Communications, 17(1–2), 7.
4. Nie, YiBin, Tan, MingJun, Liu, Gang, Ma, Lu (2020). Shortest Path Search of Expressway Internet Map Based on Improved A* Algorithm. Technology of Highway and Transport, 36(4): 110–114.

5. Xu, Nengchuang (2018). Research on the TSP Problem Based on Improved Ant Colony Algorithm. Software Guide, 17(2): 56–59.
6. Sun, Wenbin, Wang, Jiang (2016). An Algorithm for TSP Problem Based on Genetic Algorithm and Multi-optimization Operation. Geography and Geo-Information Science, 32(4): 1–4.
7. Zhao, Changxian, Fang, Muyun (2020). Design and Implementation of the Logistics Distribution System Based on Greedy Algorithm. Software Engineer, 2020, 23(5): 21–23.